Replicator dynamics and Evolutionary Stable States

Let us consider an infinite population with N types of individuals, where the fitness of an individual of type i when interacting with an individual of type j is given by A_{ij}. The replicator dynamics equation is given by:

$$\dot{x}_i = x_i[f_i(x) - \phi(x)], \tag{RE}$$

with $\phi(x) = \sum_{j=1}^{n} x_j f_j(x)$ being the average fitness in population and $f_i(x)$ being the population dependent fitness of individuals of type i

$$f_i(x) := (Ax)_i$$

The unit symplex Δ is invariant under the flow of replicator dynamics (see [**HS98**], section 7.1); in what follows we will consider its restriction to points in m-dimensional symplex $\Delta^m \subset \mathbb{R}^{m+1}$.

Definition. A point $x \in \Delta^m$ is called *Nash equilibrium* if

$$\langle x, A\hat{x}\rangle \geq \langle \hat{x}, A\hat{x}\rangle$$

and *evolutionary stable state* (ESS) if

$$\langle x, A\hat{x}\rangle > \langle \hat{x}, A\hat{x}\rangle$$

for all $\hat{x}$ in a neighbourhood of x.

Clearly ESS is a stronger condition than Nash equilibrium. A population state is evolutionary stable when all of the strategies used within the population have equal fitness and the equilibrium is stable under small perturbations. This concept has key importance in fields like behavioural ecology, evolutionary psychology, mathematical game theory and economics. ESSs are tightly related to replicator dynamics, indeed

Theorem 1 ([**HS98**], section 7.2)**.** *If $x \in \Delta^m$ is a Nash equilibrium, then it is a rest point of (RE)*

Neural Networks for rest point detections

Neural Networks provide a convenient way to detect critical points of a real valued function via gradient descent. Let us use them to detect critical points of (RE) by first define the NN architecture and loss function.

NN Architecture

i) $m+1$ neurons x_i in input layer, each of them representing the relative frequency of strategy i in the population

ii) a sequence of hidden fully connected layers

iii) $m+1$ neurons y_i in output layer, each of them representing the relative frequency of strategy i in the population.

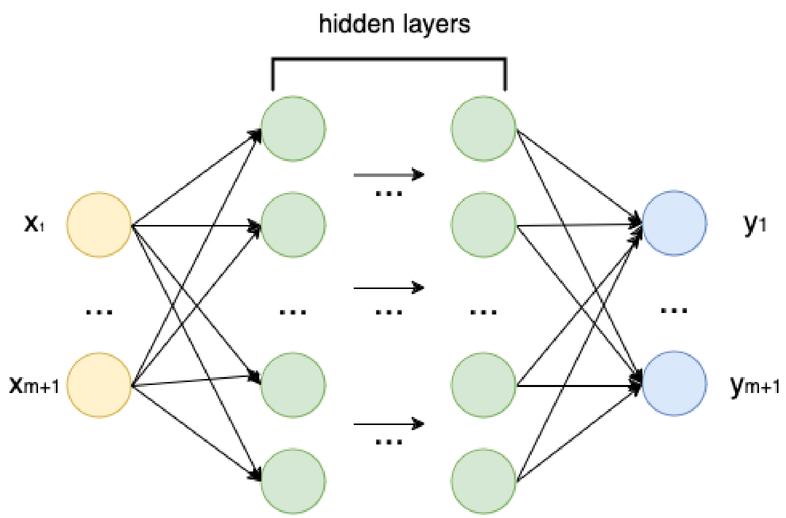

Figure 1: NN architecture

Loss

Let $x = (x_1, \ldots, x_{m+1}) \in \Delta^m$, we define a function $l : \mathbb{R}^{m+1} \to \mathbb{R}$ as

$$l(x) := \sum_{i=1}^{m+1} \sum_{j=1}^{m+1} (f_i(x) - f_j(x))^2 sign(x_i) sign(x_j) \tag{1}$$

and notice its zeros correspond to rest points of (RE), indeed x is a rest point of replicator equations iff

$$x_i = 0 \text{ or } f_i(x) - \phi(x) = 0$$

for each $i = 1, \dots, m+1$.
The function l is non negative when restricted to Δ^m, hence its zeros coincide with minima (thus critical points). At fit stage the output layer is iteratively updated to minimize the loss and we call p its representation at last training epoch. If loss vanishes on p, then p is a critical point of (RE) and then a potential ESS, since

$$\text{ESS} \subseteq \text{Nash equilibrium} \subseteq \text{rest point of (RE)}$$

ESS condition on rest points

Assuming a rest point $x \in \Delta^n$ is detected at optimization step, we wish to see if it is an ESS, by checking that x is a local minimum for the function $g_x : \Delta^m \to \mathbb{R}$ defined as

$$g_x(y) := \langle x, Ay \rangle - \langle y, Ay \rangle$$

This is done by

1) parametrizing a neighbourhood of x using a linear function $L : \mathbb{R}^m \to \mathbb{R}^{m+1}$ such that $L(0) = x$,

2) checking that 0 is a critical point of $g_x \circ L$, whose Hessian is positive definite

Assuming $x \in int(\Delta^n)$ is a rest point for (RE), the two conditions above are satisfied iff x is ESS.

Extension to Graphs

...

999

[HS98] J. Hofbauer, K. Sigmund, *Evolutionary Games and Population Dynamics.* https://api.semanticscholar.org/CorpusID:85023742, 1998.

[TJ78] Taylor, P. D, Jonker, L. B., *Evolutionarily stable states and Game Dynamics.* Mathematical Biosciences 40, 145-156. https://doi.org/10.1016/0025-5564(78)90077-9, 1978.